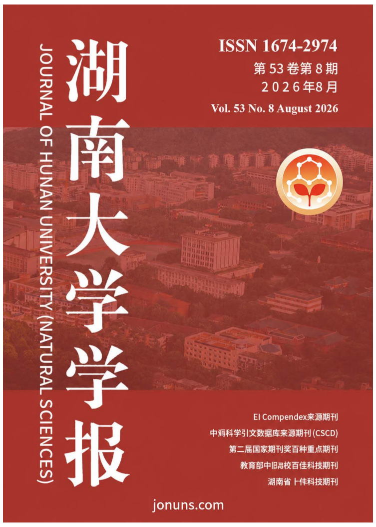

Journal of Hunan University

(Natural Sciences)

Vol. 53 No. 8

August 2026

Available online at

https://jonuns.com

ELSEVIER Scopus

Clarivate WEB OF SCIENCE

Open Access Article

https://doi.org/10.55463/issn.1674-2974.53.8.2

Cascade Control of a 2R Planar Robot for Trajectory Tracking Using an Ensemble MLP-Based Inverse Kinematics

Angie Paola Mancipe García [1], Nicolás David Barrera Fonseca [1], Jorge E Cote-Ballesteros [1], Jhon Edisson Rodríguez Castellanos [1], Fabián Barrera Prieto *[1]

[1] Department of Mechatronics, Engineering Faculty, Universidad ECCI, Bogotá, Colombia,

* Corresponding author: fbarrerap@ecci.edu.co

Article History:

Received: June 12, 2026
Revised: July 20, 2026
Accepted: August 7, 2026
Published: August 31, 2026

Abstract: This article presents the design and implementation of a cascade control scheme applied to a two degree of freedom (2R) planar robot, focused on trajectory tracking. The system architecture integrates three hierarchical loops: current, speed, and position. The current loop was characterized using the Smith two-point method, while the speed and position loops were tuned using the relay method of Åström and Hägglund, thus enabling the tuning of PI controllers. The actuators are DC motors equipped with incremental encoders for position and speed measurement, along with INA219 sensors for current measurement, programmed through the NUCLEO STM32F746ZG board. In addition, the Raspberry Pi 4 was responsible for supervision, trajectory processing, graphical interface, and neural network execution. Inverse kinematics was solved using an ensemble model of Multilayer Perceptrons (MLP) trained with synthetic data, achieving an RMSE of 0,1046° in θ_1 and 0,0533° in θ_2, outperforming individual models. The motors achieved settling times close to 1 s, overshoots below 3%, and steady-state errors under $\pm$0,12°. The system demonstrated precision in contour tracing, obtained through real-time image processing (e.g., the Windows logo), highlighting the feasibility of integrating cascade control and neural networks into low-cost embedded systems, with applications in education, automation, and robotic prototypes.

Keywords: Artificial intelligence, MLP Neural Network, Robotics, Inverse kinematics, Hierarchical control, Cascade control, Trajectory tracking, Raspberry Pi 4, STM32F746ZG.

基于集成多层感知机（MLP）逆运动学的 2R 平面机器人轨迹跟踪级联控制

摘要：本文提出了一种应用于二维自由度（2R）**平面机器人的**级联控制方案的设计与实现，重点关注轨迹跟踪。系统架构集成了三个层级控制回路：电流环、速度环和位置环。电流环采用 Smith **双点法**进行辨识，而速度环和位置环则采用 Åström **与** Hägglund **的**继电反馈法进行整定，从而实现了 PI **控制器参数的**调节。执行器采用配备增量式编码器的直流电机，用于位置与速度测量，并结合 INA219 传感器进行电流测量，所有控制均通过 NUCLEO STM32F746ZG **开**发板实现。此外，Raspberry Pi 4 负责系统监控、轨迹处理、图形界面以及神经网络的运行。

逆运动学问题通过使用基于合成数据训练的多层感知机（MLP）**集成模型求解，在** θ_1 **上**获得了 0,1046° **的** RMSE，**在** θ_2 **上**获得了 0,0533° **的** RMSE，**其性能**优于单个模型。电机实现了接近 1 s **的**稳定时间、低于 3% **的超**调量以及小于 ±0,12° **的**稳态误差。系统通过实时图像处理（例如 Windows 标志）的轮廓轨迹跟踪验证了其精度，突出了将级联控制与神经网络集成于低成本嵌入式系统中的可行性，可应用于教育、自动化以及机器人原型开发等领**域**。

.

关键词：人工智能，多层感知机（MLP）**神**经网络，机器人学，逆运动学，分层控制，级联控制，轨迹跟踪，Raspberry Pi 4，STM32F746ZG.

1. Introduction

Motion control in robotic systems is essential to ensure stable dynamic behavior in the presence of trajectory changes and disturbance rejection. Among various control strategies, the cascade control scheme is highly effective for electric motors. It enables hierarchical regulation of key variables, specifically current, speed, and position. This architecture organizes the control loops at different levels: an inner loop for current, an intermediate loop for speed, and an outer loop for position, providing a faster and more accurate response than single-loop approaches [1-3]. Furthermore, formal methods for cascade control design have been proposed. These allow for the direct calculation of gains using analytical relations based on pole placement, which ensures systematic and reproducible tuning [4].

Its application is widely documented for DC and BLDC motors. In these systems, multilevel cascade schemes employing PI/PD controllers significantly enhance dynamic performance and disturbance rejection, particularly when pulse width modulation (PWM) is used for motor driving [5-7]. Moreover, optimization techniques, such as the well-known Simulated Annealing algorithm, have been explored to tune controller parameters in brushless motor applications. This approach effectively enhances system stability, response speed, and overshoot reduction [8]. These approaches not only enhance performance but also facilitate practical implementation in contexts where manual fine-tuning is complex or poorly reproducible.

Recent studies also highlight the integration of cascade control schemes in low-cost embedded systems, particularly those implemented on STM32 microcontrollers. These devices enable the execution of current, speed, and position loops in real time, ensuring stability and low resource consumption. For instance, in [9] a low-cost PID control implementation was presented for academic environments, demonstrating that this platform provides versatile and reliable support for teaching digital control without requiring specialized hardware.

In [10], a real-time PID control scheme was implemented for a DC motor system, demonstrating stable performance and suitable dynamic response under experimental conditions. Similarly, a dual-loop fuzzy-PID speed control scheme for a DC motor was proposed in [11]. This implementation leverages the microcontroller's processing capabilities to improve adaptation to disturbances and load variations. These results confirm that the STM32 microcontrollers are efficient and flexible for the execution of advanced control algorithms, reinforcing their suitability for compact robotic applications.

Given the nonlinear dynamics, joint coupling, and sensitivity to uncertainties inherent in robotic manipulators, standalone PID schemes often fail to meet high-precision trajectory tracking requirements [12, 13]. Beyond the limitations of cascade control, the complexity of inverse kinematics, which is often hindered by computationally expensive formulations and singularity risks, further restricts traditional control methods. Consequently, this motivates the integration of artificial intelligence to enhance motion accuracy and system robustness.

2R robots are widely used as benchmarks in academic research. Their simple morphology facilitates dynamic modeling and provides an ideal platform for testing control techniques and inverse kinematics algorithms [14-16]. These manipulators require precise coordination among their actuators, such as direct current motors, in conjunction with incremental encoders that provide real-time feedback. This setup provides the necessary framework for testing the proposed cascade control scheme and assessing the performance of the MLP neural network in solving inverse kinematics.

Classical analytical or numerical methods for the inverse kinematics of serial-configuration robots show several limitations, regarding both singular positions and orientations [17-20]. Given these constraints, deep neural networks (DNNs) based on multilayer perceptron (MLPs), have emerged as effective alternatives. These networks allow for the modeling of nonlinear relationships, significantly improving accuracy in redundant manipulators compared to classical methods [21-23]. Several studies have demonstrated their application in industrial manipulators and Delta robots, achieving real-time control with low error [24, 25]. Furthermore, recent reviews highlight the potential integration of these approaches into broader planning and control frameworks that consider environments with obstacles [26].

On the other hand, recurrent neural networks (RNNs) have been employed to model SCARA robots, eliminating the need for an explicit dynamic model [27], [28]. Similarly, they have been used to control robotic prostheses based on brain signals [29]. Their applicability has even been demonstrated in embedded systems for intelligent processing tasks, such as computer vision systems with Raspberry Pi [30-33]. Nevertheless, analytical methods remain useful in unconventional configurations, such as manipulators mounted on vehicles with hybrid joints [34], or in robots with highly nonlinear dynamics such as the bicycle robot [35].

The characterization of robotic manipulators by nonlinear dynamics, joint coupling, and sensitivity to uncertainties makes precise trajectory tracking a significant engineering challenge. Although cascade control schemes are widely valued for their robustness in electric actuators [12, 13], and MLP neural networks have proven effective for solving inverse kinematics by overcoming analytical model complexities and singularities [36], [37], existing literature often addresses these aspects independently. Studies frequently focus on isolated motor control or neural-based position prediction, lacking an integrated approach suitable for real-time operation in low-cost platforms where computational efficiency is critical. Consequently, there is a clear need for a unified framework that combines lightweight control algorithms, neural-based kinematics, and machine vision within a distributed embedded architecture.

This work addresses this problem by proposing the real-time integration of a comprehensive robotic solution on a low-cost platform. The main novelty lies in the synergy of four complementary components: (i) a three-level cascade control architecture (current, speed, and position) implemented on an STM32F746ZG microcontroller [38], [39], incorporating current monitoring via INA219 sensors to prevent actuator overload [40], [41]; (ii) an ensemble of MLP neural networks trained with synthetic data for accurate inverse kinematics computation; (iii) image-based contour extraction and trajectory generation executed on a Raspberry Pi 4 [42], [43]; and (iv) a distributed embedded architecture ensuring reliable real-time communication via UART. By consolidating these techniques, this study provides a validated framework that improves trajectory tracking accuracy achieving an RMSE of 01046° for θ_1 and 0,0533° for θ_2 offering a practical and scalable solution for robotics education, rapid prototyping, and lightweight automation applications.

Finally, the article is structured as follows: Section II describes the system architecture. Section III explains the inverse kinematics model based on neural networks. Section IV details the cascade control strategy. Section V analyzes the experimental results. Section VI provides the conclusions and possible directions for future work. Section VII presents the acknowledgments for the development of this work.

2. System Architecture

Table 1 summarizes the main hardware components of the proposed robotic system, including the processing and control units, actuators, sensors, communication protocols, and their primary functions.

Table 1 Summary of the system architecture (Authors)

Category	Component	Primary function
Control	NUCLEO STM32F746ZG	Executes cascade control (current, speed, and position) for DC motors.
Processing	Raspberry Pi 4	Performs kinematic computations, machine vision

		processing, and GUI management.
Actuator	DC Motors	Drive the primary motion.
Actuator	MG90 Servomotor	Controls the vertical (Z-axis) movement.
Sensor	Incremental Encoders	Provide position and speed feedback signals to the STM32.
Sensor	INA219 Current Sensors	Monitor current consumption to prevent actuator overload.
Sensor	Camera	Captures visual information for the machine vision system.
Driver/Interface	L298 Driver	Regulates motor power via PWM signals.
Dirver/Interface	Logic-Level Converter	Adapts PWM signals to 5V levels for the MG90 servomotor.
Comunication	Bidirectional UART	Ensures real-time coordination between the Raspberry Pi and the STM32.
Comunication	VNC (Software)	Enables remote observation and control of the system from a computer.
Comunication	I2C	INA219 sensor comunication

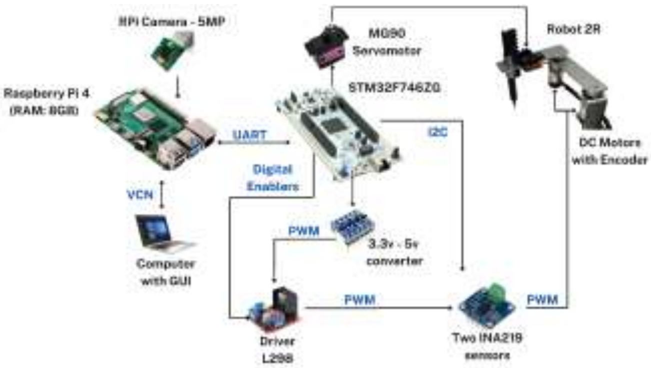

Figure 1. System interconnection (Authors)

Figure 1 and figure 2 shows the general scheme of the proposed multiplatform integration, detailing the interaction between the Raspberry Pi 4 and the STM32F746ZG board. The Raspberry Pi manages the graphical user interface, through which the operating mode is selected, and subsequently generates the reference coordinates of the trajectory that are processed by the neural network to compute the inverse kinematics (θ_1 and θ_2). The STM32F746ZG board receives these angular positions for each point along the trajectory and executes the cascade control of the DC motors, considering feedback from the sensors (encoders and INA219).

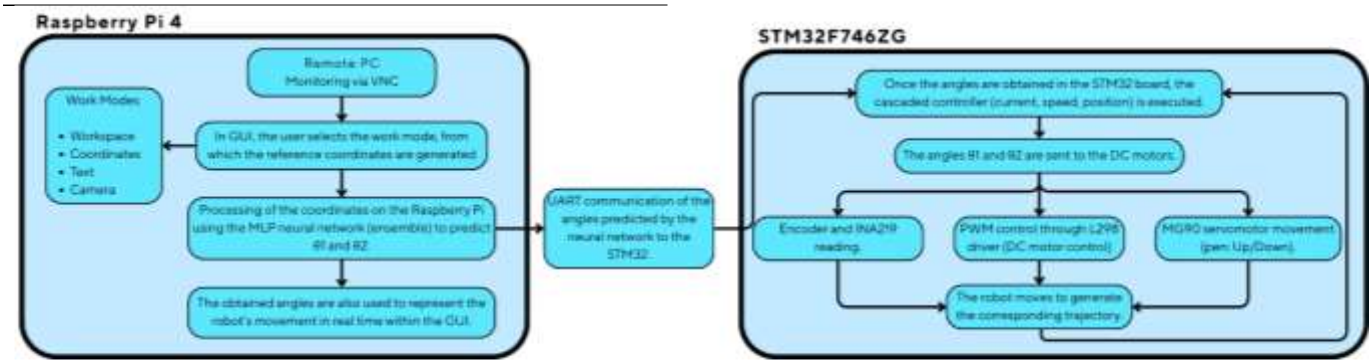

Figure 2. Logic of the multiplatform integration (Authors)

3. Inverse Kinematics Based on Artificial Neural Networks (MLP)

The neural-network-based model aims to replace the conventional computation of inverse kinematics with a learning approach using synthetic datasets. In this work, a Multilayer Perceptron (MLP) network was employed, whose structure is illustrated in Figure 3, consisting of an input layer, one or more hidden layers, and an output layer. Each neuron is connected to those in the subsequent layer, transmitting information in a single direction from input to output through weighted connections and nonlinear activation functions. This architecture enables data transformation, the learning of complex relationships between variables, and the approximation of nonlinear functions, making MLPs an effective alternative for solving the inverse kinematics of robots without relying on specific analytical models [44].

The neural network is trained using the backpropagation algorithm, which adjusts the connection weights to minimize the difference between predicted and actual values [45]. The output of each neuron is computed using Equation (1), where x_i represents the inputs, w_i the weights that determine the importance of each input, b the bias that shifts the activation function, $f(\cdot)$ the activation function that introduces nonlinearity, and y the resulting output.

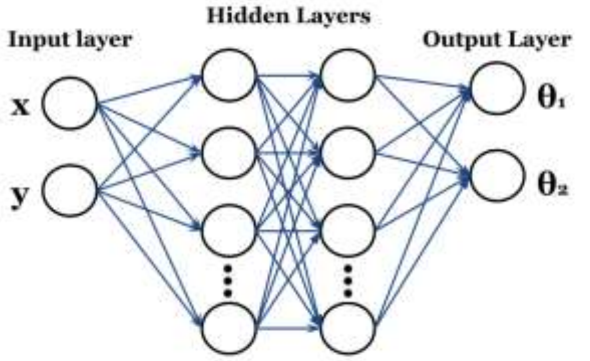

Figure 3. Architecture of the MLP Neural Network (Authors)

$$y = f\left(\sum_{i=1}^{n} x_i w_i + b\right) \quad (1)$$

For training, a dataset is generated in Google Colab using Python [46] , where the physical parameters of the robot, the dimensions of the links (l_1, l_2) and the height (h) are defined. Simulations were performed with angular variations in the range of 0 to π, with an increment of 0,001 radians; applying the Homogeneous Transformation Matrix (HTM) to obtain Cartesian positions. The dataset consists of 9.872.164 samples, which were uniformly distributed throughout the entire workspace including near-boundary configurations and excluding non-singularities [46]. The final model consists of an ensemble scheme of three MLP neural networks, taking the Cartesian coordinates (X, Y) as inputs and estimating the angles θ_1 and θ_2 as outputs. For the training process, libraries such as TensorFlow, SciPy, and scikit-learn are employed, using the Adam optimizer and the MSE loss function [47]. The key training parameters are summarized in Table 2.

Table 2 Configuration of the neural network (Authors)

Parameters	Value
Input Variables	X, Y
Output Variables	θ_1, θ_2
Training Data	70 %
Validation Data	30 %
Optimizer	Adam
Activation Function	Sigmoid
Loss Function	MSE
Error Metric	MAE

4. Cascade Control

Cascade control is a structure based on hierarchical controllers that are successively connected, considering two or more feedback loops; thus, the output of the primary (outer) controller serves as the setpoint for the secondary (inner) loop. The inner control loop must ensure the fastest dynamics in order to reject disturbances that may occur within those loops. Figure 4 shows the cascade control scheme implemented in this work, which is composed of three loops: current, speed, and position, each equipped with a PI controller in an ideal architecture.

The choice of PI controllers was based on their balance between performance and simplicity. In the inner current and speed loops, derivative action was omitted because it is not necessary. PI controllers therefore provide an appropriate compromise among fast dynamics, disturbance rejection, implementation simplicity, and zero steady-state error [48].

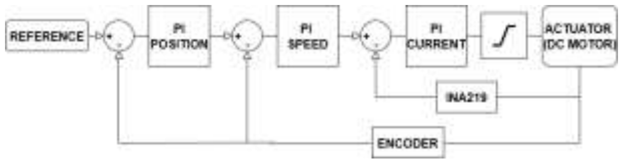

Figure 4. Block diagram of the Cascade control (Authors)

The ideal PI controller is given by Equation (2).

$$u(t) = K_p\left(e(t) + K_i \int e(t)\, dt\right) \quad (2)$$

Where:

- K_p is the proportional gain.
- K_i is the integral gain.
- $e(t)$ represents the error between the setpoint and the measured signal.

The characterization and tuning of the PI controllers for each of the current, speed, and position loops, presented below, are performed with the DC motor operating under no-load conditions.

4.1. Current Loop

4.1.1. Characterization

The INA219 current sensor was configured with a sampling period of 8 ms (125 Hz). The measured current was filtered using an Exponentially Weighted Moving Average (EWMA) filter with a smoothing factor of α = 0.005, providing effective noise attenuation while preserving the dynamic behavior required by the inner current control loop. EWMA is presented in Equation (3).

$$I_{f_k} = \alpha \cdot I_{m_k} + (1-\alpha) \cdot I_{f_{k-1}} \quad (3)$$

Where:

- I_{f_k} is the actual filtered current.
- I_{m_k} is the actual measured current.
- α represent the smoothing factor between 0 and 1.

- $I_{f_{k-1}}$ is the previous filtered current.

In Figure 5, the red line indicates the applied PWM setpoint of 100%, while the blue line represents the motor current response, which reaches 147 mA at steady state.

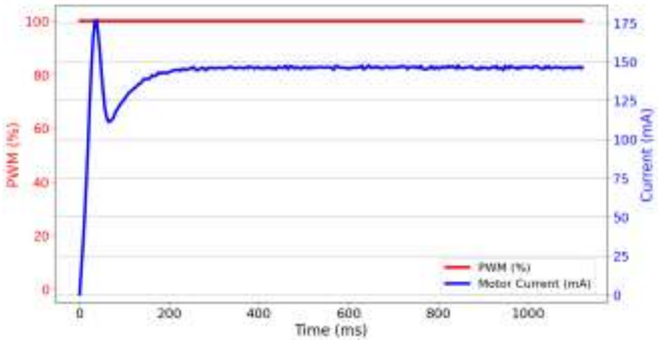

Figure 5. Motor current response (Authors)

From Figure 5, the static gain K is calculated, which is given by Equation (4).

$$K = \frac{Y_{final} - Y_{initial}}{U_{final} - u_{initial}} \quad (4)$$

$$K = 1{,}47 \text{ mA}/\%$$

The current characterization is presented in the Transfer Function (TF) of Equation (5), it was obtained using the Smith two-point method for a first-order plus dead time (FOPDT) system, where the time constant (τ) is 45,15 ms and the dead time (t_0) is 4,75 ms.

$$G(s) = \frac{1{,}47}{0{,}04515s + 1} \cdot e^{0{,}00475s} \quad (5)$$

4.1.2. Tuning

The Smith method optimized by the IAE criterion is used to tune the PI controller of the DC motor current. The proportional (K_c) and integral (K_i) gains were calculated using Equations (6) to (8), respectively.

$$K_c = \left(\frac{a_1}{K}\right)\left(\frac{t_0}{\tau}\right)^{b_1} \quad (6)$$

$$T_i = \frac{\tau}{a_2 + b_2\left(\frac{t_0}{\tau}\right)} \quad (7)$$

$$K_i = \frac{1}{T_i} \quad (8)$$

Resulting in:

- $K_c = 3{,}584\ \%$
- $T_i = 44{,}42$ ms
- $K_i = 22{,}5423\ \text{s}^{-1}$

Figure 6 shows the closed-loop response of the current controller.

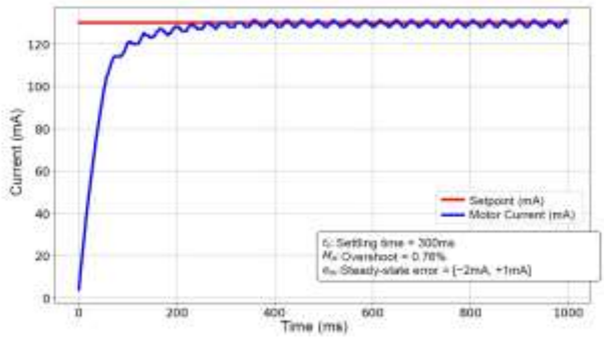

Figure 6. Closed-loop motor current response (Authors)

4.2. Speed Loop

4.2.1. Characterization

Considering that the current loop is already tuned and closed, the Relay method is employed for the characterization of speed [49, 50]. Varying current setpoints, induced sustained speed oscillations (see Figure 7).

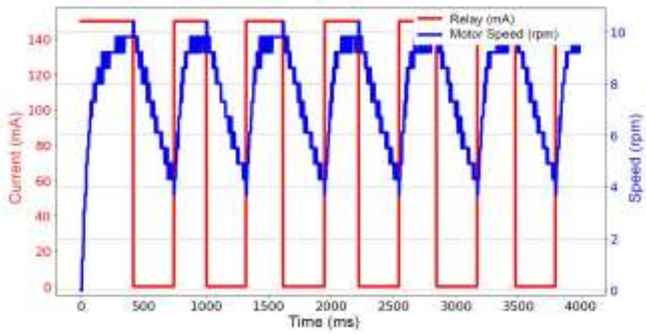

Figure 7. Motor speed response (Authors)

Based on Figure 7, the oscillation period (t_u) is 0,532 s, the average amplitude of the control signal (d) is 75 mA, the amplitude of the system response (A) is 6,7 rpm, and the ideal amplitude of the system response (ε) is 6 rpm. With these parameters, the ultimate gain (K_u) is calculated by Equation (9).

$$K_u = \frac{4d}{\pi(A^2 - \varepsilon^2)} \quad (9)$$

$$K_u = 32{,}0273 \text{ mA/rpm}$$

4.2.2. Tuning

The static gain of the system was determined through Equation (4), based on the parameters from Figure 8, in which a current step of 150 mA was applied and the achieved speed of 12,4 rpm was measured.

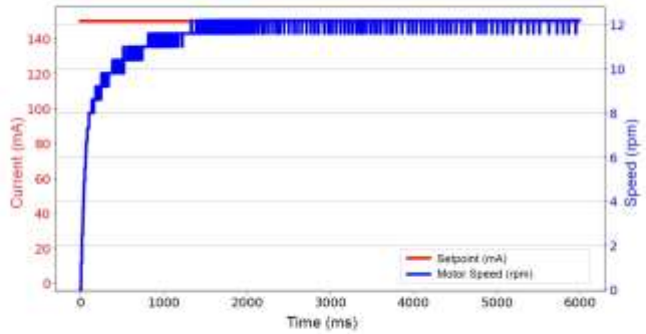

Figure 8. Motor speed response with current closed-loop (Authors)

Thus:

$$K = 0{,}0826 \text{ rpm/mA}$$

With these data, the Åström and Hägglund method [51], [52] was used to tune the proportional and integral gains of the speed PI controller, using Equations (10) and (11), respectively.

$$K_c = \frac{5K_u}{6}\left(\frac{12 + KK_u}{15 + 14KK_u}\right) \qquad (10)$$

$$T_i = \frac{t_u}{5}\left(1 + \frac{4}{15}KK_u\right) \qquad (11)$$

Resulting in:

- $K_c = 7{,}5116$ mA/rpm
- $T_i = 0{,}3315$ s
- $K_i = 3{,}0166\ \text{s}^{-1}$

Figure 9 shows the response of the speed controller with the current and speed closed-loops, highlighting a stable response.

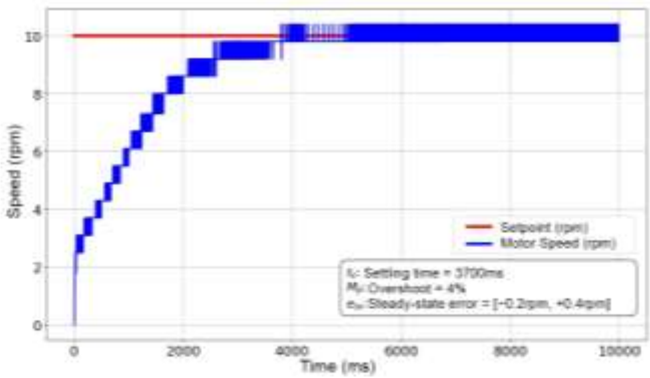

Figure 9. Closed-loop current-speed cascade (Authors)

4.3. Position Loop

4.3.1. Characterization

With the internal current and speed loops closed and tuned, the relay method was applied again to characterize the position, generating cyclic oscillations (see Figure 10).

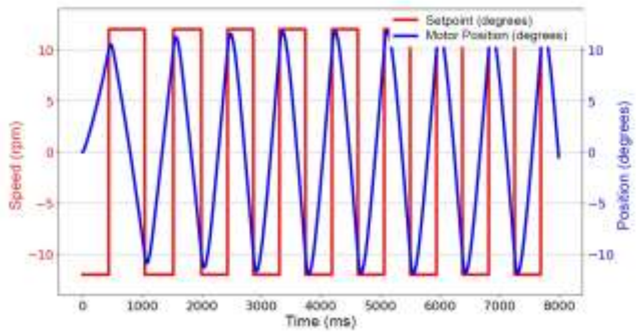

Figure 10. Motor position response (Authors)

Based on Figure 10, the oscillation period (t_u) is 0,858 s, the average amplitude of the control signal (d) is 12 rpm, the amplitude of the system response (A) is 23,46°, and the ideal amplitude of the system response (ε) is 20°. With these parameters and using Equation (9), the ultimate gain is calculated.

$$K_u = 1{,}2459 \text{ rpm/°}$$

4.3.2. Tuning

Based on the parameters from Figure 11, the static gain of the system was calculated using Equation (4). By applying a position setpoint of 50° with the current and speed loops active, and a proportional controller with $K_c = 1$ in the position loop, a stable position of 64,7° was obtained.

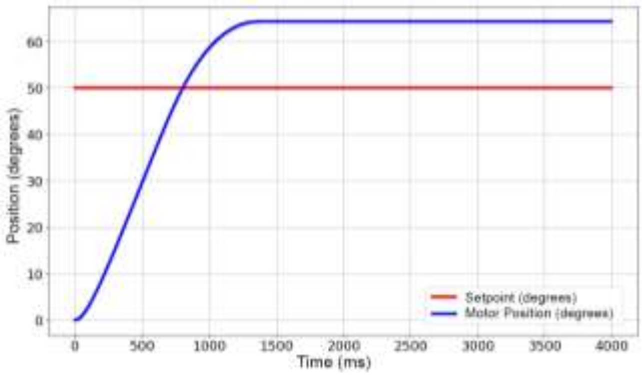

Figure 11. Motor position response with closed-loop current-speed cascade (Authors)

Thus: $$K = 1{,}294 \text{ rpm/°}$$

In the same way as in the speed loop, the Åström and Hägglund method is used to tune the proportional and integral gains of the position PI controller, using Equations (10) and (11), respectively. Resulting in:

- $K_c = 0{,}3761$ rpm/°
- $T_i = 0{,}2453$ s
- $K_i = 4{,}0766\ \text{s}^{-1}$

Figure 12 shows the response of the position controller with the current, speed, and position closed-loops, demonstrating that the response is stable.

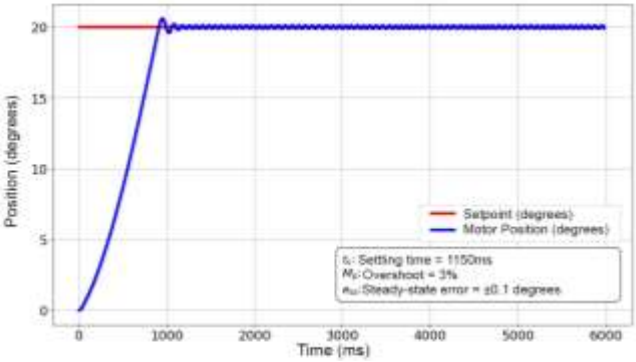

Figure 12. Closed-loop current-speed-position cascade cascade controller (Authors)

5. Results

5.1. Neural Networks Models Evaluation

Table 2 summarizes the performance of five neural network configurations. These models vary in hidden layers, neurons, and training epochs to identify the optimal structure for accuracy and stability. The proposed model achieves an RMSE of 0,01046 and 0,0533 relative to the geometric solution, validating its precision.

Therefore, the ensemble of three neural networks has reduced the variability of predictions and improved the fit compared to a single MLP [53]. Concretely, for any desired position in Cartesian coordinates (X, Y) within the robot's workspace, the model accurately predicts the required joint angles θ_1 and θ_2, effectively replacing the analytical computation of inverse kinematics.

Table 3 Validation metrics of the neural network models (Authors)

Model	Neurons	Layers	Epochs	Error (RMSE)
Model 1	200	1	128	$\theta_1 = 0,9904°$ $\theta_2 = 0,6722°$
Model 2	250	2	59	$\theta_1 = 0,2525°$ $\theta_2 = 0,1714°$
Model 3	250	3	39	$\theta_1 = 0,2495°$ $\theta_2 = 0,1253°$
Model 4	250	3	100	$\theta_1 = 0,1365°$ $\theta_2 = 0,1209°$
Ensemble Model 5.1	250	3	100	
Ensemble Model 5.2	250	2	59	$\theta_1 = 0,1046°$ $\theta_2 = 0,0533°$
Ensemble Model 5.3	100	3	39	

Model 5 was constructed as an ensemble, averaging the predictions of three previously trained neural networks to reduce fluctuation and enhance overall accuracy. To validate this approach, trajectories generated from the estimated joint angles were compared against the analytical geometric solution, as table 3 shows, using the Windows logo as a benchmark. As shown in Figure 13, the ensemble model demonstrates superior precision in tracking the reference trajectory compared to individual MLP configurations. While the full methodology regarding dataset generation, hyperparameter tuning, and ensemble construction is detailed in [46], this work adopts the validated framework to focus specifically on its real-time integration within the proposed robotic system.

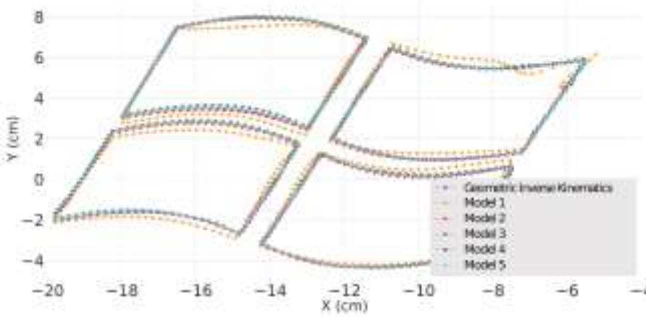

Figure 13. Comparison of geometric inverse kinematics and neural networks models trajectories (Authors)

Figure 14 presents a comparison of the angles θ_1 and θ_2 generated by the five models against those computed by analytical inverse kinematics across all 316 points of the Windows logo trajectory. The ensemble model exhibits the lowest angular deviation throughout the entire path, confirming its superior performance.

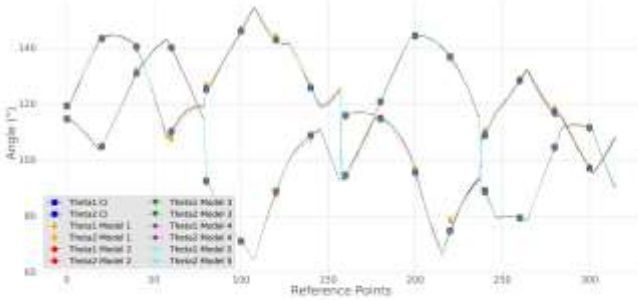

Figure 14. Comparison of geometric inverse kinematics and neural networks models angles (Authors)

5.2. Cascade Control Performance

The controller tuning procedures, based on the Smith two-point and relay methods, were performed during the no-load characterization stage for each actuator. The resulting PI parameters were then validated under load conditions, during which the controllers were subjected to the dynamic loads imposed by the manipulator links and transmission mechanisms. To validate the effectiveness of the implemented cascade control system, independent experimental tests were conducted on both joint actuators. These tests enabled the evaluation of the dynamic behavior of each loop (current, speed, and position) in terms of settling time, overshoot, and steady-state error.

5.2.1. Motor 1

Table 4 presents data from the characterization of the three loops (current, speed, and position) of motor 1 using Smith and Relay methods.

Table 4 Motor 1 characterization (Authors)

Loop	Method	Value
Current	Smith (two points)	$G(s) = \dfrac{1{,}47}{0{,}04515s + 1} \cdot e^{0{,}00475s}$
Speed	Relay	$T_u = 0{,}718$ s $d = 75$ mA $A = 6{,}7$ rpm $\varepsilon = 6$ rpm $K_u = 32{,}0273$ mA/rpm
Position	Relay	$T_u = 0{,}912$ s $d = 12$ rpm $A = 23{,}6\,°$ $\varepsilon = 20°$ $K_u = 1{,}2195$ rpm/°

Table 5 details the gains used in the cascade controller for motor 1, for each of the PI loops.

Table 5 Motor 1 tuning (Authors)

Control loop	K_c	K_i
Current	27,5349	20,5069
Speed	7,8878	4,2381
Position	0,3424	3,6593

Figure 15 shows the step response of the cascade controller for motor 1 with a step input of 20°, demonstrating the effective performance of the controller in accurately and stably reaching the setpoint.

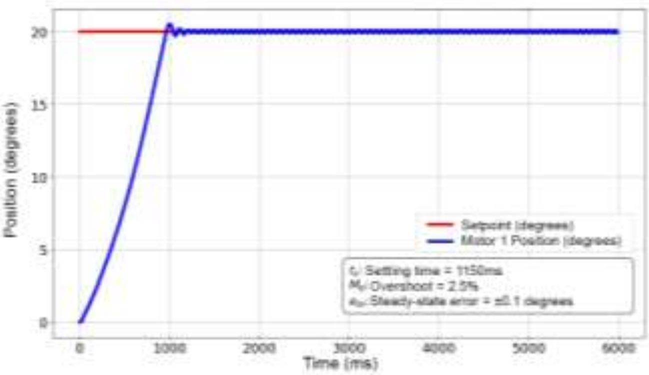

Figure 15. Response of the cascade controller for motor 1 position (Authors)

In order to evaluate the performance of the cascade controller for motor 1, performance metrics such as RMSE, IAE, and ITAE were calculated (see Table 6). All three-performance metrics are low, indicating that the cascade controller exhibits high precision (RMSE), minimal accumulated total error, meaning it corrects rapidly (IAE), and fast convergence to the setpoint with a low steady-state error (ITAE).

Table 6 Metrics for Motor 1 (Authors)

Metric	Value
RMSE	4,7184°
IAE	$1.0531x10^4$ °
ITAE	$5.1058x10^6$ °

5.2.2. Motor 2

Table 7 presents the data from the characterization of the three loops (current, speed, and position) of motor 2 using the Smith and Relay methods.

Table 7 Motor 2 characterization (Authors)

Loop	Method	Value
Current	Smith (two points)	$G(s) = \dfrac{1{,}49}{0{,}0585s + 1} \cdot e^{0{,}00458s}$
Speed	Relay	$T_u = 0{,}533$ s $d = 75$ mA $A = 6{,}7$ rpm $\varepsilon = 6$ rpm $K_u = 32{,}0273$ mA/rpm
Position	Relay	$T_u = 0{,}92$ s $d = 12$ rpm $A = 23{,}72\,°$ $\varepsilon = 20°$ $K_u = 1{,}1980$ rpm/°

Table 8 presents the gains used in the cascade controller for motor 2, for each of the PI loops.

Table 8 Motor 2 tuning (Authors)

Control loop	K_c	K_i
Current	4,5652	16,981
Speed	4,3027	3,0759
Position	0,4181	4,1374

Figure 16 shows the step response of the cascade controller for motor 2 with a step input of 20°, which, similarly to motor 1, demonstrates the effective performance of the controller in accurately and stably reaching the setpoint.

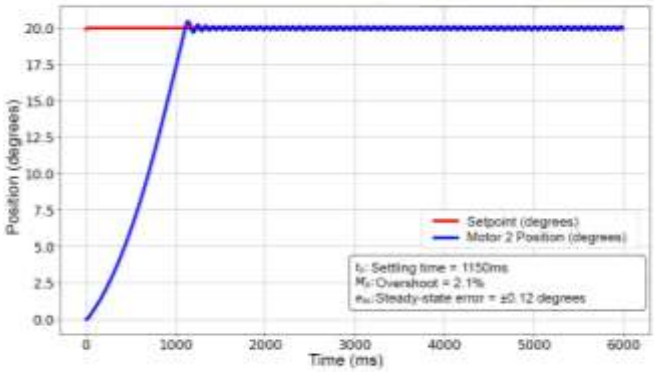

Figure 16. Response of the cascade controller for motor 2 position (Authors)

To evaluate the performance of the cascade controller for motor 2, the RMSE, IAE, and ITAE were calculated (see Table 9). As with motor 1, these three metrics are low, indicating proper controller tuning, with small deviation, fast response, and good performance in both transient and steady-state.

Table 9 Metrics for Motor 2 (Authors)

Metric	Value
RMSE	5,0435°
IAE	$1{,}1948x10^{4}$ °
ITAE	$6{,}1163x10^{6}$ °

5.3. Trajectories

Figure 17 presents the 2R robot trajectory tracking test, carried out to evaluate the performance of the cascade controllers for each motor integrated into the robotic arm. The angles generated by the neural network (θ_1 and θ_2), corresponding to the Windows logo were used. These angles were sent to the controllers to execute the 2R robot motion, and subsequently, the angles reached by the robot two joints were obtained through the motor encoders.

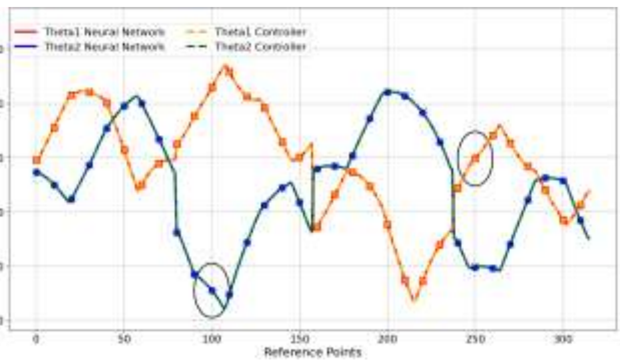

Figure 17. Response of the cascade controllers for the Windows logo trajectory (Authors)

At reference points 100 and 250 in Figure 17, a zoomed-in view was performed, as shown in Figures 18 and 19, respectively, in order to visualize and compare in greater detail the behavior of the controllers with respect to the neural network inverse kinematics.

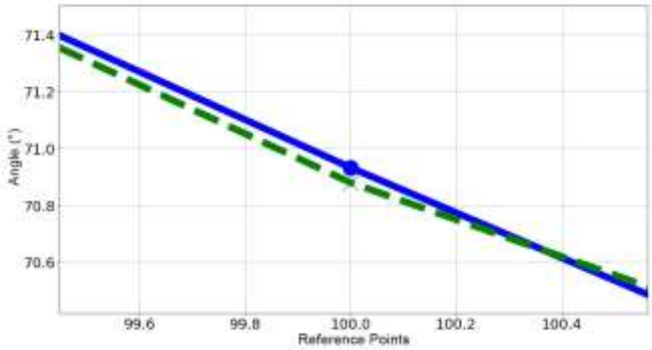

Figure 18. Zoomed-in view at reference point 100 (Authors)

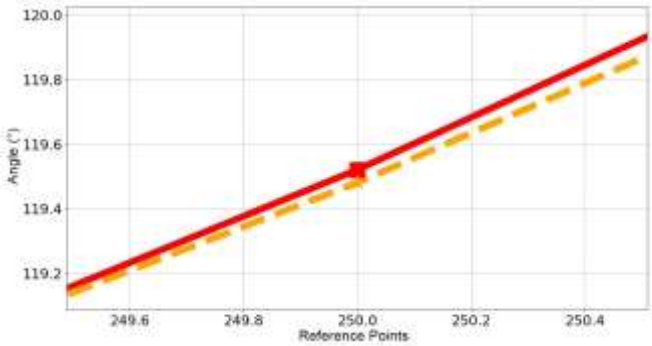

Figure 19. Zoomed-in view at reference point 250 (Authors)

To further evaluate the generalization capability of the proposed ensemble MLP-based inverse kinematics model, its performance was assessed using an independent test dataset. Table 10 summarizes the angular prediction error statistics, including the mean, minimum, maximum, and standard deviation, demonstrating a suitable accuracy and consistency of the proposed model on previously unseen data.

Table 10 Neural Network Test-Set Performance (Authors)

Metrics	Values
Mean	0,05257°
Minimum	$1{,}15967°x10^{-5}$
Maximum	0,11247°
Standard deviation	0,02706°

To complement the visual trajectory-tracking results, Table 11 presents the Root Mean Square Error (RMSE) of the two joint angles during the experimental tests. The low RMSE values obtained for both θ_1 and θ_2 confirm the well tracking accuracy achieved by the proposed cascade control and ensemble MLP-based inverse kinematics approach.

Table 11 Trajectory tracking error (Authors)

Angle	RMSE
θ_1	0,05912°
θ_2	0,07197°

5.4. Graphical User Interface and Contour Tracing

To facilitate interaction between the user and the robotic arm, a Graphical User Interface (GUI) was developed, serving as a link between information processing and the physical execution of the trajectories. Its main purpose is to allow the user to define tasks intuitively and monitor the system performance in real time.

Figure 20 shows the GUI developed in Qt Designer using Python, which integrates image capture, contour processing, trajectory visualization, and contour tracing functions. This tool serves both as an input and output peripheral, providing data related to the generated trajectory and the robot status.

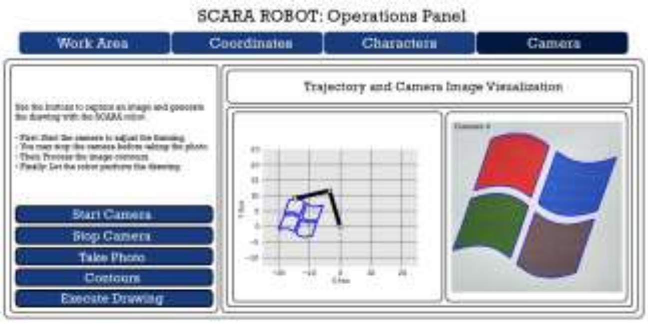

Figure 20. GUI in image capture and contour tracing mode (Authors)

Figure 21 shows the physical result obtained by the robot when tracing the contour of the Windows logo. It is evident that the cascade control scheme enabled stable trajectory tracking, maintaining continuity and smoothness in the tracing. This result validates the effectiveness of the implemented control strategy, as the trajectory defined through the neural network was successfully reproduced in the physical prototype.

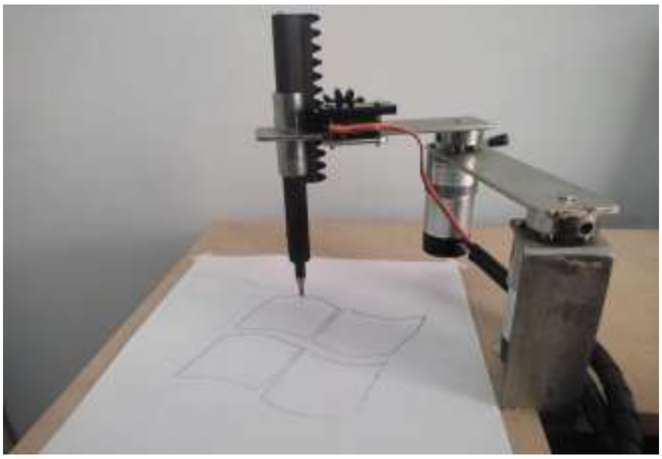

Figure 21. Result of the Windows logo tracing with the physical 2R robot (Authors)

5.5. Comparison of Actuators

Additionally, the performance of the TD-8120MG servomotor (integrated with an AS5600 encoder) and the cascade-controlled DC motor were evaluated to assess their resolution and repeatability in angular positioning. In the case of the servomotor, the relationship between pulse width and output angle was not completely linear. Dead zones were identified between 500 µs to 560 µs and 2400 µs to 2500 µs, which limited its effective range to approximately 174° instead of the theoretical 180°. The system achieved a resolution of 0,93° and a repeatability error of ±1° under identical PWM signals, revealing performance limitations attributed to friction and mechanical tolerances (Figure 22).

In the DC motor, the system response was evaluated under small increments of the angular position setpoint. It was observed that the controller maintained the steady-state error within ±0,1° for increments of 0,3°,0,2°, and 0,1°. However, when the setpoint was decreased by 0,05°, the steady-state error was affected, indicating that the actuator resolution limit under the proposed control scheme is 0,1° (Figure 23).

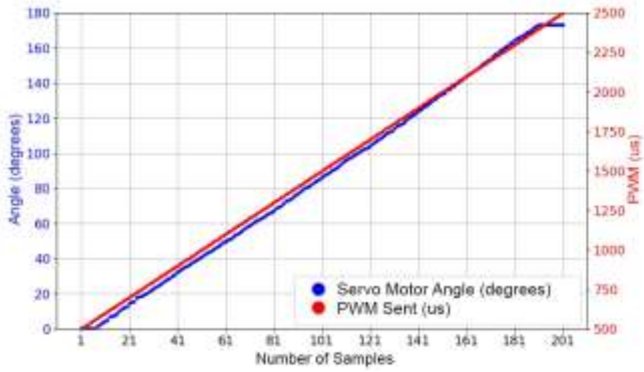

Figure 22. Comparison of the servomotor response (Authors)

This comparison underscores the necessity of the proposed cascade architecture: unlike the fixed, opaque control loops of commercial servomotors TD-8120MG, the cascade approach enables direct current regulation. This capability allows the system to dynamically adapt to varying loads and compensate for nonlinearities, effectively overcoming the inherent mechanical limitations of standard low-cost actuation.

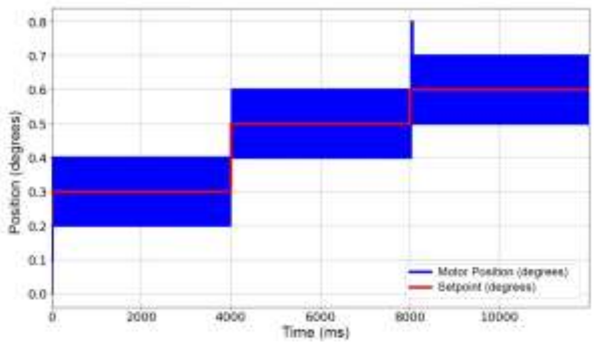

Figure 23. Motor DC response (Authors)

The comparison supports the final design choice by demonstrating that the proposed DC motor solution not only enables the implementation of the three-level cascade control architecture but also provides superior positioning performance compared with the commercial TD-8120MG servomotor. The experimental results showed better resolution and repeatability for the DC motor, confirming that the selected actuation system satisfies the accuracy requirements of the proposed trajectory-tracking application while preserving the flexibility and low-cost characteristics of the embedded robotic platform

6. Discussion

The proposed cascade control architecture is evaluated against the well-documented performance constraints of conventional PID control in low-cost robotic manipulators. By contextualizing the results within these established benchmarks, this work highlights the inherent limitations of standard PID schemes and underscores the effectiveness of the proposed hierarchical approach in enhancing trajectory tracking accuracy [12] and [13]. The implemented cascade control scheme achieved overshoots below 3%, settling times close to 1s, and steady-state errors smaller than $\pm 0{,}12°$, which suggests a stable and effective response across the tested conditions. While these results align with existing cascade control studies for DC and BLDC motors [1], [5], [6], and [18], this work advances the field by demonstrating the full integration of this architecture into a real-time 2R planar manipulator, moving beyond of the isolated actuator control.

Furthermore, the trajectory-tracking experiments provide a practical validation of the proposed cascade control architecture under load conditions. Although PI parameters were derived during no-load characterization, the system was validated under operational conditions. This final testing phase subjected the controllers to the full dynamic complexity of the robot, specifically accounting for link inertia and mechanical loading. The results indicate that the controllers maintain consistent tracking performance, with minimal settling times and steady-state errors under these conditions, suggesting that the proposed tuning strategy is effective and well-suited for practical robotic applications.

Although sophisticated control schemes like fuzzy-PID [11] and evolutionary optimization [2], [8] have been documented to enhance robotic performance, the proposed cascade architecture offers a pragmatic balance between tracking accuracy and computational simplicity, as demonstrated through the experimental characterization performed in this work. For inverse kinematics, the ensemble MLP neural network model achieved RMSE errors of $0{,}1046°$ for θ_1 and $0{,}0533°$ for θ_2. This performance aligns with, and in some cases extends, the results documented in industrial manipulators using MLP models in [21], real-time Delta robots in [24], and neural models applied to SCARA configuration in [27] and [28].

Similarly, while studies such as [17], [19], and [26] highlight the limitations of analytical and numerical methods regarding singularities and high computational costs, the proposed approach demonstrated that a neural network trained with synthetic data can generate smooth and accurate trajectories with low computational cost on embedded platforms.

From a practical perspective, the results confirm the feasibility of the system for applications in educational environments, light automation, and robotic prototyping, particularly in tasks involving trajectory tracing, visual inspection, or basic manipulation. The implementation on STM32 and Raspberry Pi embedded platforms is consistent with the trend reported in [9], [10], [30], [31], and [33], where the potential of low-cost systems to execute real-time control and computer vision algorithms is highlighted.

The assessment of the system for real-world industrial environments revealed specific operational limitations. Although initial experimental tests were conducted under controlled conditions with moderate loads, system performance may vary under external disturbances, abrupt load changes, or more complex dynamics, as noted in [3] and [38]. With a resolution close to $0{,}1°$, this level of precision may still be insufficient for high-accuracy industrial processes such as fine assembly or precision manufacturing.

7. Conclusions

The implemented cascade control scheme validates the stability and precision of hierarchical current, speed, and position loops in the 2R planar manipulator. By decoupling system dynamic regulating torque, stabilizing response, and ensuring smooth trajectory tracking this architecture overcomes the limitations inherent to independent PI configurations, such as uncontrolled current peaks, speed oscillations, and loop interference.

Experimental characterization via open-loop and relay methods enabled systematic PI tuning for each subsystem. The resulting performance, characterized by overshoots below 3%, settling times close to 1 s, and steady-state errors under $\pm 0{,}12°$, confirms the effectiveness of the proposed approach for achieving stable and repeatable trajectory tracking in low-cost robotic platforms.

For the DC motors with encoders, the cascade architecture was evaluated via performance indices for the position variable. RMSE, IAE, and ITAE values were 0.059, 0.0166 and 0.00262 for Motor 1, and 0.072, 0.0200 and 0.00313, for Motor 2, respectively. These results confirm precise, sustained angular regulation. Overall, the cascade controller significantly outperforms open-loop control and standard servomotor configurations in tracking stability and precision.

Replacing analytical inverse kinematics with an MLP ensemble for joint reference generation enhances both stability and singularity avoidance. The model achieved an RMSE of 0.1046° for θ_1 and 0.0533° for θ_2, showing superior accuracy over standalone networks. These results validate the proposed integration of cascade control and machine learning as a robust, cost-effective framework for 2R robotic position control.

The system's current performance is constrained by several technical limitations. UART communication between Raspberry Pi and STM32 introduces deterministic latency, which may affect control performance in more demanding real-time applications. Additionally, mechanical backlash and the inherent workspace limitations of the manipulator restrict positioning precision and the reachable operating region, respectively. Furthermore, the proposed inverse kinematics model was trained using a synthetic dataset, and its generalization capability under operating conditions significantly different from those represented during training has not yet been extensively evaluated.

Experimental validation was limited to functional trajectory-tracking tests and did not include comprehensive assessments under varying payloads or industrial operating conditions. Future research will focus on integrating image processing, supervision, and control into a unified ROS-based embedded system, as well as developing adaptive algorithms to compensate for mechanical nonlinearities, improve robustness under varying payloads, and further enhance the generalization capability of the proposed approach

Declarations

Author Contributions

Software, validation, investigation, data curation, writing—original draft preparation, and visualization, Angie Paola Mancipe García and Nicolás David Barrera Fonseca; conceptualization, methodology, supervision, formal analysis, and writing—review and editing, Jorge E Cote-Ballesteros, Jhon Edisson Rodríguez Castellanos and Fabián Barrera Prieto; project administration, and funding acquisition, Fabián Barrera Prieto. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement

Data available in a publicly accessible repository that does not issue DOIs: Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/NicolasB0429/Proyecto-de-grado-ROBOT-SCARA.

Funding

This research was supported by the internal call 009-2024 of Universidad ECCI through the funding of the project: Design, Construction, and Control of a 2R Planar Robotic Arm for Trajectory Tracking, ID: IN-S09-07.

Acknowledgements

The authors would like to thank the Universidad ECCI for the time and resources that made the development of this work possible.

Institutional Review Board Statement

Not Applicable.

Conflicts of Interest

The author declares that there is no conflict of interest regarding the publication of this manuscript. In addition, ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

References

[1] T. N. GÜCIN, M. BIBEROGLU, B. FINCAN, and M. O. GÜLBAHÇE, “Tuning Cascade PI(D) Controllers in PMDC Motor Drives: A Performance Comparison for Different Types of Tuning Methods,” in *2015 9th International Conference on Electrical and Electronics Engineering (ELECO)*, Bursa, Turkey: IEEE, Feb. 2016, pp. 1061–1066. doi: 10.1109/ELECO.2015.7394556.

[2] M. VEDADI, A. I. SOLIMAN, and R. KENNEL, “Optimized Angular Position Control of Brushless DC Motor Using Imperialist Competitive Algorithm Based on FOC and Trapezoidal Control,” in *2021 23rd European Conference on Power Electronics and Applications (EPE’21 ECCE Europe)*, Ghent, Belgium: IEEE, Oct. 2021, p. P.1-P.10. doi: 10.23919/EPE21ECCEEurope50061.2021.9570609.

[3] M. AFKAR, R. GAVAGSAZ-GHOACHANI, M. PHATTANASAK, S. PIERFEDERICI, and W. SAKSIRI, “Local-Stability Analysis of Cascaded Control for a Switching Power Converter,” *Journal of Robotics and Control (JRC)*, vol. 5, no. 1, pp. 160–172, 2024, doi: 10.18196/jrc.v5i1.20302.

[4] C. BUDAI, T. TÓTH-KATONA, and P. STUMPF, “Novel design method for cascade control structure of electric drives: Closed-form expressions for control gains via pole placement,” *IET Control Theory and Applications*, vol. 18, no. 17, pp. 2448–2467, Nov. 2024, doi: 10.1049/cth2.12747.

[5] M. F. CANKURTARAN and A. E. KOCAMIS, “Sensorless Speed Control of PMDC Motor with Cascade PI Controller,” in *2019 International Symposium ELMAR*, Zadar, Croatia: IEEE, Dec. 2019. doi: 10.1109/ELMAR.2019.8918654.

[6] A. MERABET, “Cascade second order sliding mode control for permanent magnet synchronous motor

drive," *Electronics (Switzerland)*, vol. 8, no. 12, pp. 1–13, Dec. 2019, doi: 10.3390/electronics8121508.

[7] D. HANDAYA and R. FAUZIAH, "Proportional-integral-derivative and linear quadratic regulator control of direct current motor position using multi-turn based on LabView," *Journal of Robotics and Control (JRC)*, vol. 2, no. 4, pp. 332–336, Jul. 2021, doi: 10.18196/jrc.24102.

[8] JAROSLAV. DUDRIK, "Brushless DC Motor Controller Optimization Using Simulated Annealing," in *2019 International Conference on Electrical Drives & Power Electronics (EDPE)*, The High Tatras, Slovakia: IEEE, Oct. 2019, pp. 292–297. doi: 10.1109/EDPE.2019.8883924.

[9] K. SOZAŃSKI, "Low Cost PID Controller for Student Digital Control Laboratory Based on Arduino or STM32 Modules," *Electronics (Switzerland)*, vol. 12, no. 15, pp. 1–23, Aug. 2023, doi: 10.3390/electronics12153235.

[10] I. SHAMTA and M. ZEYVELI, "Real-Time Implementation and Performance Evaluation of a PID-Controlled DC Motor System Using STM32," *Authorea*, pp. 1–7, May 2025, doi: 10.22541/au.174705847.75205805/v1.

[11] W. LU, "A Kind of PWM DC Motor Speed Regulation System Based on STM32 with Fuzzy-PID Dual Closed-Loop Control," in *Intelligent Computing Methodologies*, vol. 13395, Xian, China: Springer, Cham, 2022, pp. 106–113. doi: 10.1007/978-3-031-13832-4_9.

[12] D. T. TRAN, N. M. HOANG, N. H. LOC, Q. T. TRUONG, and N. T. NHA, "A Fuzzy LQR PID Control for a Two-Legged Wheel Robot with Uncertainties and Variant Height," *Journal of Robotics and Control (JRC)*, vol. 4, no. 5, pp. 612–620, Sep. 2023, doi: 10.18196/jrc.v4i5.19448.

[13] Y. K. YASEEN, A. H. MHMOOD, M. R. SUBHI, A. B. RAKAN, and H. A. MOHAMMED, "Advanced Flowrate Control of Petroleum Products in Transportation: An Optimized Modified Model Reference PID Approach," *Journal of Robotics and Control (JRC)*, vol. 4, no. 5, pp. 591–599, Sep. 2023, doi: 10.18196/jrc.v4i5.19177.

[14] N. NURNUANSUWAN, A. LONKLANG, and K. CHAMNIPRASART, "A Prototyping of 2-DOF Robot Arm Using Feedback Control System," in *2019 5th International Conference on Control, Automation and Robotics (ICCAR)*, Beijing, China: IEEE Press, Aug. 2019, pp. 368–372. doi: 10.1109/ICCAR.2019.8813726.

[15] F. J. TRIVENO VARGAS and C. ARROYO, "Modelamiento y simulación de un robot SCARA de tres grados de libertad empleando SolidWorks y Matlab/Simulink®," *Acta Nova revista de Ciencias y Tecnología*, vol. 10, no. 3, pp. 318–344, Jun. 2025, doi: 10.35319/acta-nova/202254.

[16] R. LIU, F. NAGEOTTE, P. ZANNE, M. DE MATHELIN, and B. DRESP-LANGLEY, "Deep reinforcement learning for the control of robotic manipulation: A focussed mini-review," *Robotics*, vol. 10, no. 1, pp. 1–13, Mar. 2021, doi: 10.3390/robotics10010022.

[17] W. CHOU and Y. LIU, "An Analytical Inverse Kinematics Solution with the Avoidance of Joint Limits, Singularity and the Simulation of 7-DOF Anthropomorphic Manipulators," *Transactions of FAMENA*, vol. 48, no. 1, pp. 117–132, Jan. 2024, doi: 10.21278/TOF.481055523.

[18] M. VEDADI, "Optimized Cascaded Position Control of BLDC Motors Considering Torque Ripple," in *2025 International Conference on Emerging Technologies in Electronics, Computing, and Communication (ICETECC)*, Jamshoro, Pakistan: IEEE, Apr. 2025, pp. 1–6. doi: 10.1109/ICETECC65365.2025.11071137.

[19] D. O. SANTOS, L. MOLINA, J. G. N. CARVALHO, E. A. N. CARVALHO, and E. O. FREIRE, "Modifications of Fully Resampled PSO in the Inverse Kinematics of Robot Manipulators," *IEEE Robot. Autom. Lett.*, vol. 9, no. 2, pp. 1923–1928, Feb. 2024, doi: 10.1109/LRA.2024.3349927.

[20] Y. V. THI, N. W. YAO, H. N. HUU, C. P. VAN, and T. N. MANH, "Neural Network-Based Adaptive Robust Fractional PID Control for Robotic Systems," *Journal of Robotics and Control (JRC)*, vol. 6, no. 4, pp. 1790–1801, Jul. 2025, doi: 10.18196/jrc.v6i4.26460.

[21] S. B. ŠEGOTA, N. ANĐELIĆ, V. MRZLJAK, I. LORENCIN, I. KURIC, and Z. CAR, "Utilization of multilayer perceptron for determining the inverse kinematics of an industrial robotic manipulator," *Int. J. Adv. Robot. Syst.*, vol. 18, no. 4, pp. 1–11, Jul. 2021, doi: 10.1177/1729881420925283.

[22] M. N. VU, F. BECK, M. SCHWEGEL, C. HARTL-NESIC, A. NGUYEN, and A. KUGI, "Machine learning-based framework for optimally solving the analytical inverse kinematics for redundant manipulators," *Mechatronics*, vol. 91, p. 102970, May 2023, doi: 10.1016/j.mechatronics.2023.102970.

[23] V. HLAVÁČ, "Accuracy of the Inverse Kinematics of a Planar Redundant Manipulator Solved by an MLP Neural Network," in *Advances in Mechanism Design IV*, Springer, Cham, Aug. 2024, pp. 202–211. doi: 10.1007/978-3-031-70251-8_22.

[24] A. GHOLAMI, T. HOMAYOUNI, R. EHSANI, and J. Q. SUN, "Inverse kinematic control of a delta robot using neural networks in real-time," *Robotics*, vol. 10, no. 4, pp. 1–12, Dec. 2021, doi: 10.3390/robotics10040115.

[25] F. E. AYSAL, İ. ÇELİK, E. CENGİZ, and Y. OĞUZ, "A Comparison of Multi-Layer Perceptron and Inverse Kinematic for RRR Robotic Arm," *Politeknik Dergisi*, vol. 27, no. 1, pp. 121–131, Feb. 2024, doi: 10.2339/politeknik.1092642.

[26] A. CALZADA-GARCIA, J. G. VICTORES, F. J. NARANJO-CAMPOS, and C. BALAGUER, "A

Review on Inverse Kinematics, Control and Planning for Robotic Manipulators With and Without Obstacles via Deep Neural Networks," *Algorithms*, vol. 18, no. 1, pp. 1–19, Jan. 2025, doi: 10.3390/a18010023.

[27] J. HERNÁNDEZ DÍAZ, M. IPARRAGUIRRE MONZÓN, C. MONTENEGRO CASTRO, J. ALVA ALCÁNTARA, and E. MANZANO RAMOS, "Modelado no paramétrico de robot SCARA 3-GDL utilizando redes neuronales recurrentes," *Revista de Investigaciones*, vol. 10, no. 4, pp. 379–390, Dec. 2021, doi: 10.26788/riepg.v10i4.3502.

[28] R. BOUZID, H. GRITLI, and J. NARAYAN, "ANN Approach for SCARA Robot Inverse Kinematics Solutions with Diverse Datasets and Optimisers," *Applied Computer Systems*, vol. 29, no. 1, pp. 24–34, Jun. 2024, doi: 10.2478/acss-2024-0004.

[29] M. X. GONZÁLEZ, A. M. BRUGIATI, D. H. CORNEJO, and C. I. PINZÓN, "Prototipo de mano robótica controlada mediante el procesamiento de señales cerebrales utilizando redes neuronales recurrentes," *Revista de Iniciación Científica*, vol. 6, no. 4, pp. 54–59, Jul. 2021, doi: 10.33412/rev-ric.v6.0.3154.

[30] W. GU, Z. CHANG, Q. WANG, and W. XU, "Design of intelligent robot based on Raspberry PI vision recognition," in *Ninth International Symposium on Sensors, Mechatronics, and Automation System (ISSMAS 2023)*, L. Pan and Z. Zhou, Eds., SPIE, Mar. 2024, p. 76. doi: 10.1117/12.3014817.

[31] S. E. MATHE, A. C. PAMARTHY, H. K. KONDAVEETI, and S. VAPPANGI, "A Review on Raspberry Pi and its Robotic Applications," in *2022 2nd International Conference on Artificial Intelligence and Signal Processing (AISP)*, Vijayawada, India: IEEE, Apr. 2022, pp. 1–6. doi: 10.1109/AISP53593.2022.9760590.

[32] N. RUSSO, T. MADSEN, and K. NIKOLIC, "An Implementation of Communication, Computing and Control Tasks for Neuromorphic Robotics on Conventional Low-Power CPU Hardware," *Electronics (Switzerland)*, vol. 13, no. 17, pp. 1–21, Sep. 2024, doi: 10.3390/electronics13173448.

[33] D. GRECO, M. FASIHIANY, A. V. RANJBAR, F. MASULLI, S. ROVETTA, and A. CABRI, "Computer Vision Algorithms on a Raspberry Pi 4 for Automated Depalletizing," *Algorithms*, vol. 17, no. 8, pp. 1–15, Aug. 2024, doi: 10.3390/a17080363.

[34] L. ZHENG, "Analytical Inverse Kinematics for a Prismatic-Revolute Hybrid Joints Radiography Robot Mounted on the Ambulance," *IEEE/ASME Transactions on Mechatronics*, vol. 30, no. 2, pp. 1366–1376, Aug. 2024, doi: 10.1109/TMECH.2024.3442782.

[35] M. N. HUYNH, H. N. DUONG, and V. H. NGUYEN, "A Comparative Study of Nonlinear Control and Passivity-Based Control using Neural Networks for A Bicycle Robot," *Journal of Robotics and Control (JRC)*, vol. 6, no. 2, pp. 594–604, Mar. 2025, doi: 10.18196/jrc.v6i2.26073.

[36] W. ARIBOWO, B. SUPRIANTO, and U. T. KARTINI, "Cooperation search algorithm for tuning automatic voltage regulator system," *International Journal of Power Electronics and Drive Systems*, vol. 13, no. 3, pp. 1761–1769, Jul. 2022, doi: 10.11591/ijpeds.v13.i3.pp1761-1769.

[37] E. EKER, M. KAYRI, S. EKINCI, and D. IZCI, "A New Fusion of ASO with SA Algorithm and Its Applications to MLP Training and DC Motor Speed Control," *Arab. J. Sci. Eng.*, vol. 46, no. 4, pp. 3889–3911, Jul. 2021, doi: 10.1007/s13369-020-05228-5.

[38] W. ZHAI and L. DONG, "Design and implementation of two-wheeled self-balancing mobile robot control system based on STM32-MAT and Android," *Journal of Measurements in Engineering*, vol. 13, no. 3, pp. 596–616, Jul. 2025, doi: 10.21595/jme.2025.24881.

[39] A. MAARIF and N. R. SETIAWAN, "Control of dc motor using integral state feedback and comparison with pid: Simulation and arduino implementation," *Journal of Robotics and Control (JRC)*, vol. 2, no. 5, pp. 456–461, Jul. 2021, doi: 10.18196/jrc.25122.

[40] M. RADONJIĆ, Ž. ZEČEVIĆ, and B. KRSTAJIĆ, "An IoT System for Real-Time Monitoring of DC Motor Overload," *Electronics (Switzerland)*, vol. 11, no. 10, pp. 1–14, Jul. 2022, doi: 10.3390/electronics11101555.

[41] D. A. N. K. SUHERMANTO, "Monitoring DC Motor Based on LoRa and IOT," *Journal of Robotics and Control (JRC)*, vol. 5, no. 1, pp. 54–61, Jul. 2024, doi: 10.18196/jrc.v5i1.19642.

[42] C. LOSADA-GUTIÉRREZ, F. ESPINOSA, C. CRUZ, and B. P. ALVARADO, "Electronic Control Unit and Digital Twin Based on Raspberry Pi 4 for Testing the Remote Nonlinear Trajectory Tracking of a P3-DX Robot," *Actuators*, vol. 14, no. 8, pp. 1–16, Jul. 2025, doi: 10.3390/act14080376.

[43] J. VEGA and J. M. CAÑAS, "Open vision system for low-cost robotics education," *Electronics (Switzerland)*, vol. 8, no. 11, pp. 1–20, Jul. 2019, doi: 10.3390/electronics8111295.

[44] E. A. RUELAS, J. CRUZ, E. RUELAS. IEEE, "Statistical Control of Multivariant Processes Through the Artificial Neural Network Multilayer Perceptron and the MEWMA Graphic Analysis," *IEEE Latin America Transactions*, vol. 18, no. 6, pp. 1041–1048, May 2020.

[45] S. KUNINTI and S. ROOBAN, "Backpropagation Algorithm and its Hardware Implementations: A Review," in *Journal of Physics: Conference Series*, IOP Publishing, Mar. 2021, pp. 1–10. doi: 10.1088/1742-6596/1804/1/012169.

[46] A. P. M. GARCÍA, N. D. B. FONSECA, J. E. C. BALLESTEROS, J. E. R. CASTELLANOS, and F. B. PRIETO, "Cinemática inversa para un robot planar 2R basada en una red neuronal con datos sintéticos a través de un enfoque de ensamble de modelos," *Revista colombiana de tecnologias de avanzada (RCTA)*, vol. 2,

no. 46, pp. 40–48, Jul. 2025, doi: 10.24054/rcta.v2i46.3583.
[47] Y. SHAO, F. M. DIETRICH, C. NETTELBLAD, and C. ZHANG, “Training algorithm matters for the performance of neural network potential: A case study of Adam and the Kalman filter optimizers,” *Journal of Chemical Physics*, vol. 155, no. 20, pp. 1–9, Jul. 2021, doi: 10.1063/5.0070931.
[48] K. J. . ÅSTRÖM and TORE. HÄGGLUND, *PID Controllers : theory, design, and tuning*. Wiley, 1995.
[49] K. TONG, C. GRUSSLER, and M. S. CHONG, “Self-sustained oscillations in discrete-time relay feedback systems,” in *2025 IEEE 64th Conference on Decision and Control (CDC)*, Rio de Janeiro, Brazil: IEEE, Dec. 2025, pp. 4125–4130. doi: 10.1109/CDC57313.2025.11312346.
[50] M. RABI, “Relay Self-Oscillations for Second Order, Stable, Nonminimum Phase Plants,” *IEEE Trans. Automat. Contr.*, vol. 66, no. 9, pp. 4282–4288, Oct. 2020, doi: 10.1109/tac.2020.3030893.
[51] A. REHAN, I. BOIKO, and Y. ZWEIRI, “Autotuning of PID controller using the modified relay feedback test and optimization under uncertainty principle for robotic manipulators,” *IET Control Theory and Applications*, vol. 18, no. 5, pp. 581–602, Mar. 2024, doi: 10.1049/cth2.12592.
[52] D. PAVKOVIĆ, D. LISJAK, D. KOLAR, and M. CIPEK, “PI/PID Controller Relay Experiment Auto-Tuning with Extended Kalman Filter and Second-Order Generalized Integrator as Parameter Estimators,” *Tehnicki Vjesnik*, vol. 30, no. 3, pp. 715–723, Apr. 2023, doi: 10.17559/TV-20221003112627.
[53] M. A. GANAIE, M. HU, A. K. MALIK, M. TANVEER, and P. N. SUGANTHAN, “Ensemble deep learning: A review,” *Eng. Appl. Artif. Intell.*, vol. 115, p. 105151, Oct. 2022, doi: 10.1016/j.engappai.2022.105151.

参考文献:

[1] T. N. GÜCIN, M. BIBEROGLU, B. FINCAN, and M. O. GÜLBAHÇE, “永磁直流电机（PMDC）驱动中级联 PI(D) 控制器的整定：不同整定方法性能比较,” 载于 2015年第九届电气与电子工程国际会议（ELECO）， 土耳其布尔萨：IEEE，2016年2月，第1061–1066页. doi: 10.1109/ELECO.2015.7394556.
[2] M. VEDADI, A. I. SOLIMAN, and R. KENNEL, “基于磁场定向控制（FOC）和梯形控制的帝国竞争算法无刷直流电机角位置优化控制,” 载于 2021年第23届欧洲电力电子与应用会议（EPE'21 ECCE Europe）, 比利时根特：IEEE，2021年10月，第P.1–P.10页. doi: 10.23919/EPE21ECCEEurope50061.2021.9570609.
[3] M. AFKAR, R. GAVAGSAZ-GHOACHANI, M. PHATTANASAK, S. PIERFEDERICI, and W. SAKSIRI, “开关电源变换器级联控制的局部稳定性分析,” 机器人与控制杂志（Journal of Robotics and Control，JRC）， 第5卷，第1期，第160–172页，2024年. doi: 10.18196/jrc.v5i1.20302.
[4] C. BUDAI, T. TÓTH-KATONA, and P. STUMPF, “电驱动系统级联控制结构的新型设计方法：基于极点配置的控制增益闭式表达式,” IET控制理论与应用，第18卷，第17期，第2448–2467页，2024年11月. doi: 10.1049/cth2.12747.
[5] M. F. CANKURTARAN and A. E. KOCAMIS, “采用级联 PI 控制器的 PMDC 电机无传感器速度控制,” 载于 2019年ELMAR国际研讨会, 克罗地亚扎达尔：IEEE，2019年12月. doi: 10.1109/ELMAR.2019.8918654.
[6] A. MERABET, “永磁同步电机驱动的级联二阶滑模控制,” 电子学（瑞士版）, 第8卷，第12期，第1–13页，2019年12月. doi: 10.3390/electronics8121508.
[7] D. HANDAYA and R. FAUZIAH, “基于 LabView 多圈测量的直流电机位置比例–积分–微分与线性二次调节器控制,” 机器人与控制杂志（Journal of Robotics and Control，JRC）, 第2卷，第4期，第332–336页，2021年7月. doi: 10.18196/jrc.24102.
[8] JAROSLAV DUDRIK, “采用模拟退火算法的无刷直流电机控制器优化,” 载于 2019年电气驱动与电力电子国际会议（EDPE）, 斯洛伐克高塔特拉山：IEEE，2019年10月，第292–297页. doi: 10.1109/EDPE.2019.8883924.
[9] K. SOZAŃSKI, “基于 Arduino 或 STM32 模块的低成本学生数字控制实验室 PID 控制器,” 电子学（瑞士版）, 第12卷，第15期，第1–23页，2023年8月. doi: 10.3390/electronics12153235.
[10] I. SHAMTA and M. ZEYVELI, “基于 STM32 的 PID 控制直流电机系统实时实现与性能评估,” Authorea 学术平台，第1–7页，2025年5月. doi: 10.22541/au.174705847.75205805/v1.
[11] W. LU, “一种基于 STM32 的模糊 PID 双闭环 PWM 直流电机调速系统,” 载于 智能计算方法学, 第13395卷，中国西安：Springer，Cham，2022年，第106–113页. doi: 10.1007/978-3-031-13832-4_9.
[12] D. T. TRAN, N. M. HOANG, N. H. LOC, Q. T. TRUONG, and N. T. NHA, “具有不确定性和可变高度的双轮机器人模糊 LQR-PID 控制,” 机器人与控制杂志（Journal of Robotics and Control，JRC）, 第4卷，第5期，第612–620页，2023年9月. doi: 10.18196/jrc.v4i5.19448.
[13] Y. K. YASEEN, A. H. MHMOOD, M. R. SUBHI, A. B. RAKAN, and H. A. MOHAMMED, “石油产品运输流量高级控制：一种优化的改进模型参考 PID

方法," 机器人与控制杂志（Journal of Robotics and Control，JRC），第4卷，第5期，第591–599页，2023年9月. doi: 10.18196/jrc.v4i5.19177.

[14] N. NURNUANSUWAN, A. LONKLANG, and K. CHAMNIPRASART, "采用反馈控制系统的两自由度（2-DOF）机器人机械臂原型设计," 载于 2019年第五届控制、自动化与机器人国际会议（ICCAR），中国北京：IEEE Press，2019年8月，第368–372页. doi: 10.1109/ICCAR.2019.8813726.

[15] F. J. TRIVENO VARGAS and C. ARROYO, "采用 SolidWorks 与 Matlab/Simulink® 对三自由度 SCARA 机器人进行建模与仿真," Acta Nova 科学与技术期刊, 第10卷，第3期，第318–344页，2025年6月. doi: 10.35319/acta-nova/202254.

[16] R. LIU, F. NAGEOTTE, P. ZANNE, M. DE MATHELIN, and B. DRESP-LANGLEY, "用于机器人操作控制的深度强化学习：一项重点小型综述," 机器人学, 第10卷，第1期，第1–13页，2021年3月. doi: 10.3390/robotics10010022.

[17] W. CHOU and Y. LIU, "一种避免关节极限与奇异性的解析逆运动学求解方法及其在七自由度拟人机械臂中的仿真," FAMENA 学报, 第48卷，第1期，第117–132页，2024年1月. doi: 10.21278/TOF.481055523.

[18] M. VEDADI, "考虑转矩脉动的无刷直流电机优化级联位置控制," 载于 2025年新兴电子、计算与通信技术国际会议（ICETECC），巴基斯坦贾姆肖罗：IEEE，2025年4月，第1–6页. doi: 10.1109/ICETECC65365.2025.11071137.

[19] D. O. SANTOS, L. MOLINA, J. G. N. CARVALHO, E. A. N. CARVALHO, and E. O. FREIRE, "机器人操作器逆运动学中完全重采样粒子群优化算法（PSO）的改进," IEEE机器人与自动化快报, 第9卷，第2期，第1923–1928页，2024年2月. doi: 10.1109/LRA.2024.3349927.

[20] Y. V. THI, N. W. YAO, H. N. HUU, C. P. VAN, and T. N. MANH, "基于神经网络的机器人系统自适应鲁棒分数阶 PID 控制," 机器人与控制杂志（Journal of Robotics and Control，JRC）, 第6卷，第4期，第1790–1801页，2025年7月. doi: 10.18196/jrc.v6i4.26460.

[21] S. B. ŠEGOTA, N. ANĐELIĆ, V. MRZLJAK, I. LORENCIN, I. KURIC, and Z. CAR, "利用多层感知机确定工业机器人操作器逆运动学," 国际先进机器人系统期刊, 第18卷，第4期，第1–11页，2021年7月. doi: 10.1177/1729881420925283.

[22] M. N. VU, F. BECK, M. SCHWEGEL, C. HARTL-NESIC, A. NGUYEN, and A. KUGI, "基于机器学习的冗余机械臂解析逆运动学优化求解框架," 机电一体化, 第91卷，第102970页，2023年5月. doi: 10.1016/j.mechatronics.2023.102970.

[23] V. HLAVÁČ, "采用 MLP 神经网络求解平面冗余机械臂逆运动学的精度分析," 载于 机构设计进展 IV, Springer，Cham，2024年8月，第202–211页. doi: 10.1007/978-3-031-70251-8_22.

[24] A. GHOLAMI, T. HOMAYOUNI, R. EHSANI, and J. Q. SUN, "基于神经网络的 Delta 机器人实时逆运动学控制," 机器人学, 第10卷，第4期，第1–12页，2021年12月. doi: 10.3390/robotics10040115.

[25] F. E. AYSAL, İ. ÇELİK, E. CENGİZ, and Y. OĞUZ, "多层感知机与逆运动学在 RRR 机器人机械臂中的比较," 理工学院期刊（Politeknik Dergisi），第27卷，第1期，第121–131页，2024年2月. doi: 10.2339/politeknik.1092642.

[26] A. CALZADA-GARCIA, J. G. VICTORES, F. J. NARANJO-CAMPOS, and C. BALAGUER, "基于深度神经网络的机器人操作器逆运动学、控制与路径规划综述：含障碍与无障碍环境," 算法, 第18卷，第1期，第1–19页，2025年1月. doi: 10.3390/a18010023.

[27] J. HERNÁNDEZ DÍAZ, M. IPARRAGUIRRE MONZÓN, C. MONTENEGRO CASTRO, J. ALVA ALCÁNTARA, and E. MANZANO RAMOS, "基于循环神经网络的三自由度 SCARA 机器人非参数建模," 研究期刊（Revista de Investigaciones），第10卷，第4期，第379–390页，2021年12月. doi: 10.26788/riepg.v10i4.3502.

[28] R. BOUZID, H. GRITLI, and J. NARAYAN, "采用不同数据集与优化算法的人工神经网络（ANN）SCARA 机器人逆运动学求解方法," 应用计算机系统, 第29卷，第1期，第24–34页，2024年6月. doi: 10.2478/acss-2024-0004.

[29] M. X. GONZÁLEZ, A. M. BRUGIATI, D. H. CORNEJO, and C. I. PINZÓN, "基于循环神经网络和脑电信号处理的机器人手原型," 科学启蒙期刊, 第6卷，第4期，第54–59页，2021年7月. doi: 10.33412/rev-ric.v6.0.3154.

[30] W. GU, Z. CHANG, Q. WANG, and W. XU, "基于 Raspberry Pi 视觉识别的智能机器人设计," 载于 第九届传感器、机电一体化与自动化系统国际研讨会（ISSMAS 2023），SPIE，2024年3月，第76页. doi: 10.1117/12.3014817.

[31] S. E. MATHE, A. C. PAMARTHY, H. K. KONDAVEETI, and S. VAPPANGI, "Raspberry Pi 及其机器人应用综述," 载于 2022年第二届人工智能与信号处理国际会议（AISP），印度维杰亚瓦达：IEEE，2022年4月，第1–6页. doi: 10.1109/AISP53593.2022.9760590.

[32] N. RUSSO, T. MADSEN, and K. NIKOLIC, "基于传统低功耗 CPU 硬件的神经形态机器人通信、计算与控制任务实现," 电子学（瑞士版），第13卷，第17

期，第1–21页，2024年9月. doi: 10.3390/electronics13173448.

[33] D. GRECO, M. FASIHIANY, A. V. RANJBAR, F. MASULLI, S. ROVETTA, and A. CABRI, “基于 Raspberry Pi 4 的自动拆垛计算机视觉算法,” 算法, 第17卷，第8期，第1–15页，2024年8月. doi: 10.3390/a17080363.

[34] L. ZHENG, *et al*., “安装于救护车上的棱柱–转动混合关节放射成像机器人解析逆运动学,” IEEE/ASME 机电一体化汇刊，第30卷，第2期，第1366–1376页，2024年8月. doi: 10.1109/TMECH.2024.3442782.

[35] M. N. HUYNH, H. N. DUONG, and V. H. NGUYEN, “基于神经网络的自行车机器人非线性控制与无源控制比较研究,” 机器人与控制杂志（Journal of Robotics and Control，JRC）, 第6卷，第2期，第594–604页，2025年3月. doi: 10.18196/jrc.v6i2.26073.

[36] W. ARIBOWO, B. SUPRIANTO, and U. T. KARTINI, “用于自动电压调节器系统整定的协同搜索算法,” 国际电力电子与驱动系统期刊, 第13卷，第3期，第1761–1769页，2022年7月. doi: 10.11591/ijpeds.v13.i3.pp1761-1769.

[37] E. EKER, M. KAYRI, S. EKINCI, and D. IZCI, “ASO 与 SA 算法的新型融合及其在 MLP 训练和直流电机速度控制中的应用,” 阿拉伯科学与工程期刊, 第46卷，第4期，第3889–3911页，2021年7月. doi: 10.1007/s13369-020-05228-5.

[38] W. ZHAI and L. DONG, “基于 STM32-MAT 与 Android 的双轮自平衡移动机器人控制系统设计与实现,” 工程测量期刊, 第13卷，第3期，第596–616页，2025年7月. doi: 10.21595/jme.2025.24881.

[39] A. MAARIF and N. R. SETIAWAN, “基于积分状态反馈的直流电机控制及其与 PID 控制的比较：仿真与 Arduino 实现,” 机器人与控制杂志（Journal of Robotics and Control，JRC），第2卷，第5期，第456–461页，2021年7月. doi: 10.18196/jrc.25122.

[40] M. RADONJIĆ, Ž. ZEČEVIĆ, and B. KRSTAJIĆ, “用于直流电机过载实时监测的物联网系统,” 电子学（瑞士版），第11卷，第10期，第1–14页，2022年7月. doi: 10.3390/electronics11101555.

[41] D. A. N. K. SUHERMANTO, *et al*., “基于 LoRa 与物联网（IoT）的直流电机监测,” 机器人与控制杂志（Journal of Robotics and Control，JRC）, 第5卷，第1期，第54–61页，2024年1月. doi: 10.18196/jrc.v5i1.19642.

[42] C. LOSADA-GUTIÉRREZ, F. ESPINOSA, C. CRUZ, and B. P. ALVARADO, “基于 Raspberry Pi 4 的电子控制单元与数字孪生，用于 P3-DX 机器人远程非线性轨迹跟踪测试,” 执行器, 第14卷，第8期，第1–16页，2025年8月. doi: 10.3390/act14080376.

[43] J. VEGA and J. M. CAÑAS, “面向低成本机器人教育的开放视觉系统,” 电子学（瑞士版），第8卷，第11期，第1–20页，2019年11月. doi: 10.3390/electronics8111295.

[44] E. A. RUELAS, J. CRUZ, E. RUELAS, and D. IEEE, “基于人工神经网络多层感知机与 MEWMA 图分析的多变量过程统计控制,” IEEE 拉丁美洲汇刊, 第18卷，第6期，第1041–1048页，2020年5月.

[45] S. KUNINTI and S. ROOBAN, “反向传播算法及其硬件实现：综述,” 载于 物理学会议系列，IOP Publishing，2021年3月，第1–10页. doi: 10.1088/1742-6596/1804/1/012169.

[46] A. P. M. GARCÍA, N. D. B. FONSECA, J. E. C. BALLESTEROS, J. E. R. CASTELLANOS, and F. B. PRIETO, “基于合成数据和模型集成方法的 2R 平面机器人逆运动学神经网络研究,” 哥伦比亚先进技术期刊（RCTA）, 第2卷，第46期，第40–48页，2025年7月. doi: 10.24054/rcta.v2i46.3583.

[47] Y. SHAO, F. M. DIETRICH, C. NETTELBLAD, and C. ZHANG, “训练算法对神经网络势能模型性能的重要性：Adam 与卡尔曼滤波优化器案例研究,” 化学物理学期刊，第155卷，第20期，第1–9页，2021年11月. doi: 10.1063/5.0070931.

[48] K. J. ÅSTRÖM and T. HÄGGLUND, PID 控制器：理论、设计与整定. Wiley，1995.

[49] K. TONG, C. GRUSSLER, and M. S. CHONG, “离散时间继电反馈系统中的自维持振荡,” 载于 2025 年第 64 届 IEEE 决策与控制会议（CDC）, 巴西里约热内卢：IEEE，2025年12月，第4125–4130页. doi: 10.1109/CDC57313.2025.11312346.

[50] M. RABI, “二阶稳定非最小相位系统中的继电器自激振荡,” IEEE 自动控制汇刊, 第66卷，第9期，第4282–4288页，2020年10月. doi: 10.1109/TAC.2020.3030893.

[51] A. REHAN, I. BOIKO, and Y. ZWEIRI, “采用改进继电反馈试验和不确定性原理优化的机器人操作器 PID 控制器自动整定,” IET 控制理论与应用, 第18卷，第5期，第581–602页，2024年3月. doi: 10.1049/cth2.12592.

[52] D. PAVKOVIĆ, D. LISJAK, D. KOLAR, and M. CIPEK, “采用扩展卡尔曼滤波器和二阶广义积分器作为参数估计器的 PI/PID 控制器继电实验自动整定,” 技术公报（Tehnički Vjesnik）, 第30卷，第3期，第715–723页，2023年4月. doi: 10.17559/TV-20221003112627.

[53] M. A. GANAIE, M. HU, A. K. MALIK, M. TANVEER, and P. N. SUGANTHAN, “集成深度学习：综述,” 工程中的人工智能应用，第115卷，第105151页，2022年10月. doi: 10.1016/j.engappai.2022.105151.

Manuscript Information
Word count: 11,944 words (excluding references).
Peer-Review Record

Fast-track status: Not fast-tracked.
First-round reviews received: 3 reports.
Revision cycles completed: 3 rounds.
Final version submitted: August 7, 2026

Disclaimer / Publisher's Note

The statements, opinions, and data contained in this article are solely those of the authors and do not necessarily represent the views of the Journal of Hunan University (Natural Sciences) or its editorial team. The journal and its editors disclaim any responsibility for injury to persons or property resulting from any ideas, methods, instructions, or products referred to in the content of this article.